REMARKS

Applicants respectfully request reconsideration. Claims 1-8 were previously pending in this application. By this amendment, claims 13, 16, 19, 26, and 27 have been canceled. Claims 1, 14, 15, 17, 18, 20, and 22-24 have been amended. Claims 9-15, 17, 18, and 20-25 were previously withdrawn. New claims 28 and 29 have been added. As a result, claims 1-8, 28, and 29 are pending for examination, with claims 1, 28 and 29 being independent.

Rejections Under 35 U.S.C. §101

Claims 1 through 8 are rejected under 35 U.S.C. §101 because the Office Action asserts that the claim language is directed to non-statutory subject matter." Claims 1 is amended to refer to a system comprising computer-executable modules encoded on a computer-storage medium. Accordingly, withdrawal of the rejection of claims 1 through 8 under 35 U.S.C. §101 is respectfully requested.

Rejections Under 35 U.S.C. §103

Claims 1 through 8 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,959,439 (Boike) in view of Applicant Admitted Prior Art (AAPA, Applicants "Background" Disclosure of U.S. Patent Publication 2005/0090248). Applicants respectfully disagree.

Before discussing the claims, Applicant provides a brief overview of some embodiments of the invention to assist the Examiner in appreciating various aspects of the present invention. The Boike reference is also briefly discussed.

Brief Overview of Some Embodiments

Some embodiments are directed to a system for providing standard mechanisms to interface with Wireless Wide Area Network (WWAN) services. The system includes a roaming service that provides applications access to information about the status and capabilities of the WWAN network interface as well as other notifications (¶46).

The roaming service employs a WWAN module to request information from a corresponding WWAN device driver (¶47). The commands from the WWAN module are passed through a user-mode to kernel-mode driver interface (e.g., NDIS) to the WWAN device driver (¶49). The WWAN device driver has an information base in which objects, such as configuration information and statistical information, are stored. These objects are referred to via object identifiers (OIDs). When the roaming service is to query or set an object in the WWAN device driver's information base, the roaming service provides the specific OID from a first set of OIDs for that object (¶50). Similarly, when the WWAN device driver is to proactively send data to the roaming service, an OID from a second set of OIDs is passed from the WWAN device driver with a data structure (¶52).

The interface described in the present application also includes a timing component. The roaming service may send SET and QUERY commands from a first set of OIDs to the device driver asynchronously over the user-mode to kernel-mode driver interface (¶59). When the WWAN device driver receives such a command, it typically processes the request and quickly returns a completion response in the form of an INDICATE command from a second set of OIDs (¶59).

However, some commands such as scanning for visible operators or reading entries from a Subscriber Identity Module (SIM), may take significant time to complete (e.g., seconds) (¶59). Thus, the user-mode to kernel-mode driver interface is managed to ensure commands are successfully received by the WWAN device driver despite variations in the time it takes the driver to perform operations. Initially, when an OID is received by the WWAN device driver, it immediately provides a return operation to the roaming service. The return operation indicates the command has been successfully received unless there is an immediate error condition. The WWAN device driver then performs the operation and returns a completion response (¶60).

Until the return operation is received, the user-mode to kernel-mode driver interface is blocked, preventing the roaming service from sending OIDs. After the return operation the interface is opened and the roaming service may send further OIDs. If the WWAN device driver receives an OID after sending a return operation but before sending a completion response (i.e., while it is performing operations for an existing request) it keeps track of the request by storing its RequestID in the driver's information base (¶60).

It should be appreciated that the foregoing discussion of embodiments of the invention is provided merely to assist the Examiner in appreciating various aspects of the present invention. However, not all of the description provided above necessarily applies to each of the independent claims pending in the application. Therefore, the Examiner is requested to not rely upon the foregoing summary in interpreting any of the claims or in determining whether they patentably distinguish over the prior art of record, but rather is requested to rely only upon the language of the claims themselves and the arguments specifically related thereto provided below.

Brief Overview of Boike

Boike describes a system interface abstraction layer (SIAL) that employs a standard interface between software modules and a miniport driver (col. 2, lines 13-16) but is silent on managing the timing of commands sent over the interface. The SIAL serves as a set of function calls between message source and destination modules (col. 1, lines 66-67). The SIAL includes an external interface, an internal interface, and a platform interface (col. 3, lines 51-53). The external interface interfaces with the operating system (col. 3, lines 56-60). When the operating system wishes to send a message to the miniport driver the external interface processes the message and sends it to the internal interface (col. 5, lines 1-4). The internal interface then routes the message to the miniport driver (internal entity 303) (col. 4, lines 10-12). The platform interface includes public interfaces for platform specific operations such as startup, shutdown, and initialization (col. 4, lines 53-65).

Function calls for allowing the internal entities to bind a function to a specific message, send a message to an external entity, or broadcast a message to all internal drivers are defined (col. 7, line 12 – col. line 19). Similarly, the public functions of the platform interface are described (col. 11 line 20 – col. 12, line 36).

Claim 1

Claim 1 as amended, is directed to a system for using a user-mode module to kernel-mode driver interface to send commands to and receive information from a kernel-mode wireless wide area network (WWAN) device driver. Claim 1 recites, *inter alia*, "after sending the OID from the

first set, the user-mode entity is permitted to send further OIDs from the first set to the WWAN device driver through the user-mode module to kernel-mode driver interface, prior to receiving a response from the WWAN device driver."

This limitation clearly distinguishes over the cited references. The Examiner has cited several passages of Boike as purportedly teaching these features. However, none do. The first passage, column 3 lines 6-32, generally describes the architecture of Boike's system as including a user mode, a kernel mode, and the NDIS architecture. The second passage, column 4 lines 53-65, describe the platform interface as including public interfaces for performing platform specific operations. For example, establishing and destroying connections with the operating system. Finally the third passage, column 7 lines 30-60, describes the parameters of the function SmSysIfAddMessageHandler. SmSysIfAddMessageHandler is used to "bind a call back function in an internal driver module to an occurrence of a message on [a] specified message channel" (col. 7, lines 18-20). These passages provide no description of managing the timing of commands sent over the interface and specifically fail to teach or suggest "after sending the OID from the first set, the user-mode entity is permitted to send further OIDs from the first set to the WWAN device driver through the user-mode module to kernel-mode driver interface, prior to receiving a response from the WWAN device driver."

The secondary reference does not teach or suggest the limitations not met by Boike. Thus, even if combined, the references would not teach or suggest all the limitations of the claim.

Accordingly, claim 1 patentably distinguishes over the prior art of record, so that the rejection of claim 1 under 35 U.S.C. §103 should be withdrawn.

Claims 2-8 depend from claim 1, incorporate all of its limits, and should be allowed for at least the same reasons. Though Applicants do not necessarily concur with the rejections, Applicants believe it is unnecessary to separately address the rejections of the dependent claims. However, the dependent claims also add limitations that further distinguish over the references, and Applicant reserves the right to argue further for the patentability of these claims.

New Claims 28 and 29

Independent claims 28 and 29 are added to further define Applicant's contribution to the art. Each of the independent claims recite limitations directed to managing the timing of commands sent over the interface.

For example, claim 28, which is directed to a computer-storage medium, recites "a user-mode entity configured to send a plurality of object identifiers (OIDs) from a first set of OIDs by way of the user-mode module to the kernel-mode WWAN device driver through the user-mode / kernel-mode interface, wherein, the kernel-mode WWAN device driver is configured perform an operation requested by a first received OID from the plurality of OIDs, and, when a second received OID from the plurality of OIDs is received before completion of the operation, store an identifier of the second received OID in an information base."

As another example, claim 29 recites "sending the second OID by way of the user-mode entity to the WWAN device driver through the user-mode / kernel-mode interface before receiving a third OID from a second set of OIDs from the WWAN device driver."

It should be clear from the discussion above that the prior art of record fails to satisfy these limitations. Accordingly, claims 28 and 29 patentably distinguish over the prior art of record and allowance is respectfully requested.

CONCLUSION

A Notice of Allowance is respectfully requested. The Examiner is requested to call the undersigned at the telephone number listed below if this communication does not place the case in condition for allowance.

If this response is not considered timely filed and if a request for an extension of time is otherwise absent, Applicant hereby requests any necessary extension of time. If there is a fee occasioned by this response, including an extension fee, the Director is hereby authorized to charge any deficiency or credit any overpayment in the fees filed, asserted to be filed or which should have been filed herewith to our Deposit Account No. 23/2825, under Docket No. M1103.70138US01.

Dated: September 12, 2008

Respectfully submitted,

By s/Edmund J. Walsh/
Edmund J. Walsh
Registration No.: 32,950
WOLF, GREENFIELD & SACKS, P.C.
Federal Reserve Plaza
600 Atlantic Avenue
Boston, Massachusetts 02210-2206
617.646.8000

x09/12/08x